Delphi

บุญเลิศ เอี่ยมทัศนา

Delphi 2.0 กับอินเทอร์เน็ต ตอน กับ Windows 95

เนื้อหาของบทความในชุดนี้ ตั้งใจจะให้เป็นเรื่องของเดลไฟ กับอินเทอร์เน็ต แต่ก็มีเรื่องพาดพิง มาถึง LAN ด้วย ในฉบับนี้จึงเป็น เรื่องของวินโดวส์ 95 โดยตรง

ท้าวความ

ผมได้นำโครงงานจริงมายกเป็นตัวอย่างในฉบับก่อน โดยสรุปคือมีเครื่องแม่หนึ่งเครื่องกับเครื่องลูกหลายเครื่อง โดยที่

บนเครื่องแม่มีโปรแกรมอยู่ 2 ตัว (2 แอพพลิเคชัน) ตัวแรกจะรับข้อมูลมาจากตลาดหลักทรัพย์ มาให้แก่โปรแกรมตัวที่สองเพื่อส่งให้แก่เครื่องลูก จึงมีการติดต่อเพื่อส่งข้อมูลระหว่างแอพพลิเคชันอยู่ 2 วิธีคือ ภายในเครื่องเดียวกันและต่างเครื่องกัน

กับวินโดวส์ 3.11

กับวินโดวส์ 3.11 มีทางเลือกซึ่งไม่มากนัก ดังนี้

1. ภายในเครื่องเดียวกัน อาจใช้วิธี

ส่งข้อมูล ด้วย SendMessage เมื่อทดลองใช้พบว่าช้าไป

- ทาง DDE ช้าเช่นกัน

- ใช้หน่วยความจำร่วม เป็นวิธีที่เลือกใช้ ทั้งเร็ว ทั้งง่าย โดยที่โปรแกรมแรกเพียงแค่จองหน่วยความจำแล้วส่งแอดเดรสไปให้แก่โปรแกรมตัวที่สอง เช่นด้วย SendMessage (ครั้งเดียว) โปรแกรมตัวที่สองสามารถใช้หน่วยความจำนี้ได้เช่นเดียวกับที่เป็นของตนเอง

2. ต่างเครื่องกัน ตอนนั้นเลือกใช้ NDDE (Net Dynamic Data Exchange) เพราะผมยังไม่รู้จัก TCP/IP เคยบอกว่าจะนำมาเล่าสู่กันฟัง คงไม่แล้วครับ

ของใหม่ในวินโดวส์ 95

ขอขัดจังหวะด้วยของใหม่ในวินโดวส์ 95 ก่อนครับ ซึ่งมีอยู่มากด้วยกัน ขอนำมากล่าวเพียง 2 ประเด็นก่อน

1. ประเด็นแรก ในวินโดวส์ 95 เป็นการยากที่จะแยกหน่วยความจำใน RAM กับในดิสก์ และรวมไปถึงไฟล์ด้วย เช่น เมื่อจองหน่วยความจำด้วย GetMem หรือ GlobalAlloc จะไม่ทราบครับว่าจองหน่วยความจำได้ที่ใด ใน RAM หรือในดิสก์

2. ประเด็นที่ 2 มีเรื่องอยู่ 3 เรื่องที่เกี่ยวพันกัน โดยเฉพาะ 2 เรื่องหลังคือ

2.1 Process and Threads ในคำอธิบายของวินโดวส์ 3.x จะอ้างถึง application เป็นหลัก แต่ในวินโดวส์ 95 เปลี่ยนเป็น process แทน ที่ผมจับความได้คือ ในแอพพลิเคชันหนึ่ง ๆ อาจมีได้หลาย process แต่ส่วนใหญ่จะมีเพียง process เดียว

ซึ่งใน process หนึ่ง ๆ อาจไม่มีหรือไม่เช่นนั้นต้องมีหลาย threads เพื่อทำงานในแบบ time slice ดูได้จากโปรแกรมตัวอย่างของเดลไฟในไดเรคทอรี

```
\demos\threads
```

2.2 File Mapping เพื่อแม็พไฟล์ให้เรามองเห็นเหมือนหน่วยความจำผืนหนึ่ง เช่น ถ้าเป็นเท็กซ์ไฟล์สามารถใช้ PChar ชี้ได้

2.3 Pipe โดยเฉพาะ Named Pipe เพื่อใช้เป็นท่อสำหรับการรับส่งข้อมูลระหว่าง process ทั้งในเครื่องเดียวกันและต่างเครื่อง

มาถึงวินโดวส์ 95

พอมาถึงวินโดวส์ 95 โปรแกรมเก่าใช้ไม่ได้ กล่าวคือ

1. การใช้หน่วยความจำร่วม ให้ใช้ File Mapping แทน ซึ่งถ้าเปิดไฟล์โดยไม่ให้ชื่อไฟล์ จะเท่ากับเป็นการจองหน่วยความจำ เพื่อใช้เป็นหน่วยความจำร่วม เรียกว่า Shared Memory ซึ่งไม่ยาก แต่ต้องเขียนโปรแกรมยาวกว่าหลายบรรทัด

2. การติดต่อระหว่างเครื่อง ไม่มี NDDE สามารถเลือกดำเนินการได้ดังนี้

2.1 ใช้ TCP ใน MS ActiveX ซึ่งเป็นการติดต่อผ่านทาง TCP/IP ตามที่ผมนำมากล่าวในฉบับก่อน

2.2 ใช้ Winsock ซึ่งเป็นการติดต่อผ่านทาง TCP/IP เช่นกัน ตามที่จะกล่าวในฉบับนี้

2.3 ใช้ Named Pipe

ทั้งสามเรื่องนี้ Named Pipe น่าจะเร็วที่สุด ติดขัดอยู่ที่ผมยังเขียนโปรแกรมไม่ได้ (หลายเรื่องนักต้องขอเรื่องง่าย ๆ ก่อนครับ)

โปรแกรมในชุด SockV3

ในไฟล์ sockv3.zip ตามที่ผมฝากไปกับซอฟต์เมล์ในฉบับที่ 132 มีไฟล์ดังนี้

1. sockets.pas และ sockets.dpr เป็นไฟล์ของยูนิต Sockets เพื่อทำเป็นคอมโปเนนต์ของสกุล TSockets ขอให้ย้ายไปไว้ในไดเรคทอรีที่เตรียมไว้สำหรับยูนิตชนิดนี้ แล้วทำเป็นคอมโปเนนต์โดยเลือกรายการ

```
Component/Install/Add/Browse
```

แล้วเลือกชื่อไฟล์ sockets.pas แต่ก่อนที่จะติดตั้งขอให้ดูในหัวข้อ "ปรับปรุง TSockets" ก่อนครับ

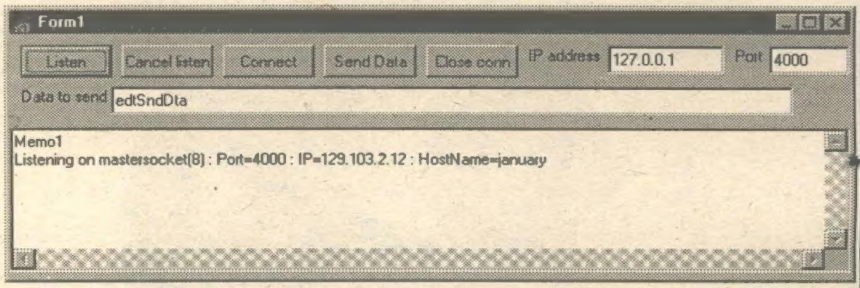
2. โปรแกรมตัวอย่างใช้ Sockets หลายโปรแกรม ที่ผมจะนำมากล่าวถึงคือ ClntServ (clntserv.*) ซึ่งอยู่ในไฟล์ cltsvr.zip อีกทอดหนึ่ง

โปรแกรมทั้งหมดนี้เขียนโดย Gary T. Desrosiers (Poland) ซึ่งเจ้าของอนุญาตให้นำไปใช้ได้ตามต้องการ และที่ผมจะนำโปรแกรม ClntServ มากล่าวถึง ผมต้องขอเขียนเป็นโปรแกรมใหม่ตามความถนัดของผม ให้ชื่อโปรแกรมเป็น ClntSrv และจะให้คำอธิบายตามโปรแกรมใหม่นี้

ปรับปรุง TSockets

ความจริงผมไม่ต้องการปรับปรุงโปรแกรมเก่า แต่เพื่อให้สามารถส่งข้อมูลได้มากกว่า 255 ไบต์ ขอให้แก้โปรแกรมของกิจกรรม TSockets.SetText ในไฟล์ sockets.pas ให้เป็นดังในลิสติงที่ 1

รูปที่ 1



ลิสติงที่ 1

```
procedure TSockets.SetText(Text: string);
var  BytesSent,len : integer; P: PChar;  txt : String;
begin
  len := Length(Text);  txt := Text;
  P := PChar(@txt[1]);
  if FMode = 0 then  SetTimeout;
  BytesSent := send(FSocket,P^,len,0);
  if FMode = 0 then  ResetTimeout;
  if BytesSent < 0 then
    SocketError(FSocket,'SetText (Send)',WSAGetLastError);
end;
```

ลิสติงที่ 2

```
procedure TForm1.EnaBtt(bLstn,bCncl,bCnnct,bSnd,
                                        bClse : Boolean);
begin
  bttLstn.Enabled := bLstn;
  bttCnclLstn.Enabled := bCncl;
  bttClseCnn.Enabled := bCncl;
  bttCnnct.Enabled := bCnnct;
  bttSndDta.Enabled := bSnd;
  bttClseCnn.Enabled := bClse;
end;
```

→

โปรแกรม ClntSrv

โปรแกรม ClntSrv ตามลิสติงที่ 2 และผลการรันเมื่อเริ่มแรกดังรูปที่ 1 เป็นโปรแกรมตัวอย่างแสดงการใช้ Sockets ในเรื่องการติดต่อระหว่างเครื่อง โดยให้เป็นได้ทั้ง Server และ Client กล่าวคือ

การรันทดสอบ

จะขอนำผลการรันโปรแกรมมากล่าวก่อนเป็นการแนะนำโปรแกรมคือ จะต้องรันโปรแกรมนี้ 2 ก๊อบปี้ เพื่อใช้ตัวหนึ่งเป็น Server และตัวหนึ่งเป็น Client ซึ่งอาจให้อยู่บนเครื่องเดียวกันหรือต่างเครื่องกัน (ครั้งแรกควรให้อยู่บนเครื่องเดียวกัน) กล่าวคือ

1. รันโปรแกรมที่จะใช้เป็น Server แล้วเลือกปุ่ม Listen จะแสดงข้อความดังในรูปที่ 1 แจ้งค่า IP address, HostName และ Port

2. รันโปรแกรมที่จะใช้เป็น Client แล้วนำค่าของ Server มาป้อนให้แก่กรอบอีดิตของ Client ดังนี้

- ค่า Port คือ 4000 แก่กรอบ Port (แต่จะมีค่าตรงกันอยู่แล้ว)

- ค่า IP คือ 129.103.2.12 หรือค่า HostName คือ january แก่กรอบ IP address (ถ้าอยู่บนเครื่องเดียวกันไม่จำเป็นต้องป้อน)

แล้วเลือกปุ่ม Connect จะแสดงข้อความแจ้งว่าติดต่อได้ (หรือไม่ได้)

3. ป้อนข้อความที่กรอบ Data to send แล้วเลือกปุ่ม Send Data จะให้ Server หรือ Client เป็นผู้ส่งได้ทั้งคู่ ข้อความนี้จะไปปรากฏในอีกหน้าต่างหนึ่ง และถ้า Server เป็นผู้รับจะส่งข้อความนั้นกลับไป (เพียงแค่เพื่อแสดงการเขียนโปรแกรมเท่านั้น)

เมื่อรันโปรแกรมตามที่กล่าวมานี้อาจมีปัญหา ขอให้อ่านหัวข้อ "ปัญหากับ TCP/IP"

เตรียมโปรแกรม

ขอให้เตรียมโปรแกรมตามรูปที่ 1 และลิสติงที่ 2 ดังนี้

1. กำหนดปุ่ม Button รวม 5 ปุ่มโดย

- ให้ชื่อปุ่ม (Caption) เป็น Listen, Cancel Listen, Connect, Send Data และ Close Conn

- ให้ชื่อออบเจ็กต์ (Name) เป็น bltLstn, bltCnclLstn, bltCnnct, blnSndDta และ bltClseCnn

ลิสติงที่ 2 (ต่อ)

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  edtPrt.Text := Sock1.Port;
  edtIpAdr.Text := Sock1.IPAddr;
  EnaBtt(True,False,True,False,False);
  iClntSrvrG := 0;
end;

procedure TForm1.bttLstnClick(Sender: TObject);
var  txtS : String;
begin
  Sock1.Port := edtPrt.Text;
  Sock1.SListen;
  EnaBtt(False,True,False,False,False);
  iClntSrvrG := 2;
  txtS := 'Listening on mastersocket(';
  txtS := txtS + IntToStr(Sock1.MasterSocket) + ') : ';
  txtS := txtS + 'Port=' +
              Sock1.GetPort(Sock1.MasterSocket) + ' : ';
  txtS := txtS + 'IP=' + Sock1.GetLocalIPAddr + ' : ';
  txtS := txtS + 'HostName=' + Sock1.HostName;
  Memo1.Lines.Add(txtS);
end;

procedure TForm1.bttCnnctClick(Sender: TObject);
begin
  Sock1.IPAddr := edtIpAdr.Text;
  Sock1.Port := edtPrt.Text;
  Sock1.SConnect;
  iClntSrvrG := 1;
  EnaBtt(False,False,False,False,false);
end;

procedure TForm1.Sock1SessionConnected(Sender: TObject;
                                       Socket: Integer);
var  txtS : String;
begin
 EnaBtt(False,False,False,TRue,True);
 txtS := 'Connected to IP=';
 txtS := txtS + Sock1.GetPeerIPAddr(sock1.SocketNumber);
 txtS := txtS + ' on his port #';
 txtS := txtS + Sock1.GetPeerPort(Sock1.SocketNumber);
 Memo1.LineS.Add(txtS);
end;


procedure TForm1.Sock1SessionAvailable(Sender: TObject;
                                       Socket: Integer);
var  new_client : TSocket;  txtS : String;
begin
  new_client := Sock1.SAccept;
  Memo1.Lines.Add('Client session accepted, socket(' +
      IntToStr(new_client) + ')' +
    ' Peer IP=' + Sock1.GetPeerIPAddr(new_client) +
    ' Peer Port='+Sock1.GetPeerPort(new_client));
  bttSndDta.Enabled := True;
end;

(* ---------------------- *)

procedure TForm1.bttSndDtaClick(Sender: TObject);
begin
  Sock1.Text := edtSndDta.Text;
end;

procedure TForm1.Sock1DataAvailable(Sender: TObject;
                                    Socket: Integer);
var  bufS : String;
begin                                                  →
```

2. กำหนดกรอบเลเบิลเป็นป้ายชื่อของกรอบอีดิต 3 คู่คือ

- กรอบ Label ให้ชื่อ (Caption) เป็น IP address, Port และ Data to send

- กรอบ Edit ให้ชื่อออบเจ็กต์ (Name) เป็น edtIpAdr, edtPrt และ edtSndDta

3. กำหนดกรอบ Memo1

4. กำหนดคอมโปเนนต์เป็น Sock1 (อยู่ในหน้า Samples) แล้วให้ค่าพรอพเพอร์ตี้ส์ดังนี้

- IP Addr เป็น 127.0.0.1 (เป็นค่าตามของเดิม)

- Port เป็น 4000 (ดูเหมือนค่าที่ให้ได้ ตั้งแต่ 1025 ถึง 4999 ไม่ยืนยันครับ จำไม่ได้ว่าอ่านพบที่ใดและตัวเลขด้วย)

ป้อนโปรแกรมตามลิสติงที่ 2

ลิสติงที่ 2 (ต่อ)

```
  Memo1.Lines.Add('Data recevied from socket(' +
                                IntToStr(Socket) + ')');
  Sock1.SocketNumber := Socket;
  bufS := Sock1.Text;
  Memo1.Lines.Add('->' + bufS + '<-');
  If iClntSrvrG = 2 Then
  begin
    Sock1.Text := bufS;
    Memo1.Lines.Add('...Buffer echoed back to client');
  end;
end;

(* ---------------- *)

procedure TForm1.bttCnclLstnClick(Sender: TObject);
begin
  if Sock1.SocketNumber <> INVALID_SOCKET then
    Sock1.SClose;
  if Sock1.MasterSocket <> INVALID_SOCKET then
    Sock1.SCancelListen;
  EnaBtt(True,False,True,False,False);
  Memo1.Lines.Add('Cancel Listen by Host');
end;

procedure TForm1.bttClseCnnClick(Sender: TObject);
begin
  Sock1.SClose;
  if iClntSrvrG <> 2 then {not server?}
  begin  EnaBtt(True,False,True,False,False);
         iClntSrvrG := 0  end;
end;

procedure TForm1.Sock1SessionClosed(Sender: TObject;
                                    Socket: Integer);
begin
  Memo1.Lines.Add('Lost connection on socket(' +
                                IntToStr(Socket) + ')');
  Sock1.SocketNumber := Socket;
  bttClseCnnClick(Self);
  bttSndDta.Enabled := False;
end;

end.
```

คำอธิบายโปรแกรม

จะขอให้คำอธิบายโปรแกรมทั้งสองภาคควบคู่กันไป ตามชื่อกิจกรรมคือ

1. bltLstnClick เมื่อเลือกปุ่ม Listen ในหน้าต่าง Server ได้ให้

- นำค่าในกรอบอีดิตมาให้เป็นค่า Sock1.Port เพื่อให้เปลี่ยนหมายเลขพอร์ตได้ในขณะรัน

- ให้เป็น Server โดยเรียกใช้กิจกรรม Sock1.SListen

โปรแกรมนอกนั้นเป็นการให้แสดงค่าต่าง ๆ ดังเช่นในรูปที่ 1 และของกิจกรรมต่อไปด้วย ซึ่งจะของดกล่าว

2. bltCnnctClick เมื่อเลือกปุ่ม Connect ในหน้าต่าง Client ได้ให้

- ค่าในกรอบอีดิตแก่ Sock1.IPAddr และ Socket1.Port

- ให้เชื่อมต่อกับ Server ด้วย Sock1.Connect

3. Sock1SessionConnected

4. Sock1SessionAvailable

จากผลของกิจกรรม Connect จะทำให้เกิดอีเวนต์ทั้งสองนี้ตามลำดับ ซึ่งเพียงแต่ให้นำค่าต่าง ๆ ออกแสดง (TSocket เป็น Integer)

5. bltSndDtaClick เมื่อเลือกปุ่ม Send Data ได้ให้ส่งข้อมูลในกรอบ Data to send ด้วย Sock1.Text จะทำให้เกิดอีเวนต์ OnData Available ในอีกหน้าต่างหนึ่ง

6. Sock1DataAvailable ในการรับข้อมูลได้ให้ดำเนินการดังนี้

- ให้ค่า Socket ของผู้ส่งแก่ SocketNumber

- ให้รับด้วยการให้ค่า Sock1.Text แก่ bufS เพื่อให้นำออกแสดง

- ถ้าผู้รับเป็น Server ให้ส่งข้อมูลกลับไปด้วย

7. bltCnclLstnClick เมื่อเลือกปุ่ม Cancel Listen ที่ Server ซึ่ง

- SClose หมายถึงถ้ามีการติดต่อให้ยุติการติดต่อ

- SCancelListen ให้ยกเลิกการ Listen

8. bltClseCnnClick เมื่อเลือกปุ่ม Close Conn ที่ Client ได้ให้ยุติการติดต่อด้วย SClose

9. Sock1SessionClosed เมื่อฝ่ายหนึ่งฝ่ายใดเรียกใช้กิจกรรม SClose จะทำให้เกิดอีเวนต์นี้ที่อีกฝ่ายหนึ่ง

ปัญหากับ TCP/IP

เมื่อรันโปรแกรมนี้แล้วอาจติดต่อกันไม่ได้ ตามที่ผมกล่าวในตอนก่อนว่า คงเห็นว่าง่ายจึงไม่บอก จากเอกสารต่าง ๆ สำหรับผมแล้วดูเหมือนจะเป็นความลึกลับที่คนทั้งโลกเขารู้กัน ส่วนผมไม่รู้อยู่คนเดียว

ตัวปัญหา

ผมค้นจนอ่อนใจจึงนึกได้ว่า ผู้ที่ขอให้ผมเขียนโปรแกรมเคยเล่าถึงโปรแกรมเกมที่เล่นระหว่างเครื่อง ซึ่งผมเคยพบว่าโปรแกรมประเภทนี้ชอบติดตั้งอะไรต่ออะไรให้ จึงขอให้เขานำไปทดลองรันดู ปรากฏว่าใช้ได้ ได้ข้อมูลมาทำให้พอทราบได้ว่าตัวปัญหาเกิดขึ้นจาก TCP/IP ซึ่งอาจแบ่งเป็น

1. ไม่ได้ติดตั้ง TCP/IP ซึ่งไม่น่าจะมี
2. ติดตั้ง TCP/IP ไว้สำหรับอินเทอร์เน็ต ดังเช่นกรณีของผม
3. เซตค่า TCP/IP ไม่ถูกต้อง

การติดตั้ง TCP/IP

ผมขอเรียนให้ทราบก่อนว่าการติดตั้ง TCP/IP ตามที่ผมจะกล่าวต่อไปนี้ จะมีทั้งการติดตั้งและถอดถอน เครื่องผมทำได้แต่เครื่องของคุณอาจมีปัญหาข้างเคียง เพราะติดตั้งซอฟต์แวร์ไว้ไม่เหมือนกัน หากไม่มั่นใจควรให้ผู้รู้ดำเนินการให้

อีกประเด็นหนึ่งคือเครื่องของผมทั้งคู่ติดตั้ง H/D ถอดเปลี่ยนได้ที่ไดรฟ์ C และผมมี H/D อยู่หลายแผ่น

สำหรับค่าต่าง ๆ ที่ผมจะนำมากล่าวเป็นค่าของเครื่องผมอาจไม่ตรงกัน เมื่อแน่ใจแล้วเริ่มเลยครับโดย เลือกการติดตั้ง (และถอดถอน) Network โดย

```
/Start /Settings /Control Panel //Network
```

(เครื่องหมาย / หมายถึงเลือก // หมายถึงดับเบิลคลิ๊ก)

จะแสดงฟอร์มเพื่อการนี้ ให้ดำเนินการต่อดังนี้ โดยอาจเลือกทำตั้งแต่ข้อ 4 ก่อน หากยังติดต่อไม่ได้จึงย้อนกลับมาทำตั้งแต่ข้อ 1 ใหม่ คือ

1. ถอนโปรโตคอลเดิมทั้งหมด โดย

```
/NE 2000 Compatible /Remove
```

2. ติดตั้งโปรโตคอลใหม่ทั้งหมด โดย

```
/Add /Adapter /Add
/(Novell/Anthen) { in left List box }
/NE 2000 Compatible { in right List box }
/OK
```

3. ถอนโปรโตคอลที่ไม่ใช้และอาจทำให้เกิดปัญหา คือ

```
3.1 /Client for Netware Network /Remove
3.2 /(IPX/SPX Compatble Protocol)  /Remove
```

4. เซตค่า NE 2000 โดย

```
/NE 2000 Compatible /Properties /Resource
Interrupt(IRQ) = 5
I/O address range = 300-31F
```

ค่าทั้งสองนี้เป็นค่าเฉพาะเครื่องซึ่งต้องทราบ

5. เซตค่า TCP/IP สำหรับการติดต่อระหว่างเครื่อง โดย

```
/(TCP/IP)  /Properties /IP Address
      /check: Specify an IP address
      IP Address: = 129.103.2.12
/OK
```

ค่า IP Address ตามแต่จะกำหนด แต่ตัวเลข 3 กลุ่มแรกตามตัวอย่างนี้คือ 129.103.2 ควรเหมือนกัน และกลุ่มสุดท้ายคือ 12 จะต้องไม่ซ้ำกันในแต่ละเครื่อง (สำหรับค่า Subnet Mark: ไม่ต้องให้ ระบบจะให้ภายหลัง)

6. ให้ใช้ไฟล์และเครื่องพิมพ์ร่วมหากต้องการ โดย

```
/File and Print Sharing
      /check: I want ... my files.
      /check: I want ... my printer(s).
/OK
```

เสร็จแล้วให้ออกจากการติดตั้ง Network โดยเลือกปุ่ม OK แล้วให้บูตเครื่องใหม่

ผลที่ได้

ทำให้สามารถติดต่อระหว่างเครื่องได้ แต่ก็ทำให้ติดต่อกับ Internet ไม่ได้

ActiveX

มีเรื่องของ MS ActiveX เพิ่มครับคือ ถึงต้นเดือนกรกฎาคม 39 ไมโครซอฟต์ได้ออกโปรแกรมที่เกี่ยวข้องอยู่กับ MS ActiveX มา 4 ตัวด้วยกันคือ

1. ActiveX Internet Controls เพื่อให้นำไปใช้ทำคอมโปเนนต์ในภาษาต่าง ๆ ดังที่ผมนำมาแนะนำในฉบับก่อน เช่นใช้ HTML ทำเป็น Web browser เพื่อดูเอกสารใน Web page

2. ActiveX Control Pad คงต้องของเรียกว่าเป็น HTML เอดิเตอร์ เพื่อสร้าง Web page ซึ่งไม่ใช่เรื่องสำคัญ ที่สำคัญคือไมโครซอฟต์ทำให้สามารถกำหนดคอมโปเนนต์เป็นคอนโทรลมาตรฐานของวินโดวส์ได้ โดยที่ใน ActiveX Control Pad มี HTML Layout Control อยู่ด้วย

3. HTML Layout Control เพื่อใช้สร้างเลย์เอาต์ ในการสร้าง Web page

4. ActiveX Component เพื่อดาวน์โหลดให้แก่โปรแกรมที่ใช้แสดงขีดความสามารถของ ActiveX Control Pad ซึ่งต้องระวังครับ เพราะเพียงแต่เลือกชื่อเท่านั้นจะดาวน์โหลดและผนวกเข้ากับระบบของวินโดวส์โดยตรง และเมื่อบังเอิญเลือกต้องรอชั่วครู่จึงจะทราบว่ากำลังดาวน์โหลด

เพียงแต่แจ้งให้ทราบครับ

Newlook Place

ขอโฮมเพจของผมครับ สำหรับเดือนกรกฎาคม 39 คือ

ของเก่า

1. โปรแกรมใช้ ActiveX ตามที่กล่าวในฉบับที่แล้วคือ

- TTCP เพื่อการติดต่อระหว่างเครื่อง
- THTML ใช้ทำเวบเพจ

2. โปรแกรมตอบคำถาม

- BrwPict แสดงภาพ (bmp, ico และ wmj) หลาย ๆ ภาพพร้อมกันในกรอบ DrawGrid เพื่อแสดงการ Drag and Drop หลายวิธี
- DirDir แสดงไดเรคทอรีพร้อมด้วยขอดรวมขนาดของไฟล์ในแต่ละไดเรคทอรี เพื่อแสดงการเขียนโปรแกรม Recursion

ของใหม่

ซึ่งคาดว่าจะนำไปไว้ก่อน 10 ก.ค. 39 คือ

- ใช้ FTP ใน ActiveX

สนใจเยี่ยมชมและดาวน์โหลดได้ที่

```
http://www.inet.co.th/cybermall/newlook
```

ท้ายตอน

ในฉบับหน้าจะนำเรื่องของ FTP ใน ActiveX มากล่าวถึง และถ้าเป็นไปได้จะนำ Component Pact ของ NetMasters (ชื่อบริษัท) ที่โฆษณาไว้ว่าดีกว่า ActiveX มากล่าวถึงครับ

อินเทอร์เน็ต
สำหรับผู้เริ่มต้น

รหัสสั่งซื้อ 5124222
ราคา 240 บาท

พาคุณเข้าสู่โลกแห่งการใช้อินเทอร์เน็ต ด้วยเนื้อหาการใช้งานโปรแกรมเกี่ยวกับอินเทอร์เน็ตที่ควรรู้อย่างละเอียด พร้อมทั้งเทคนิคที่น่าสนใจ ให้คุณเรียนรู้การใช้อินเทอร์เน็ตอย่างสมบูรณ์แบบได้ด้วยตนเอง

หาซื้อได้ตามร้านหนังสือชั้นนำทั่วประเทศ หรือที่ ซีเอ็ดบุ๊คเซ็นเตอร์ สาขาฟอร์จูนทาวน์ (ชั้นใต้ดิน) สาขาสีลมคอมเพล็กซ์ (ชั้น 4) สาขาหลักสี่พลาซ่า (ชั้น 3) สาขาอิมพีเรียลเวิลด์สำโรง (ชั้นใต้ดิน) สาขาเซ็นทรัลซิตี้บางนา (ชั้น 3) สาขาเดอะมอลล์ บางกะปิ บางแค (ชั้น G) สาขาฟิวเจอร์ปาร์ค บางแค (ชั้นใต้ดิน) และสามารถสั่งซื้อทางไปรษณีย์โดยตรงที่ บริษัท ซีเอ็ดยูเคชั่น จำกัด (มหาชน) 800/43-45 ซ.ตระกูลสุข ถ.อโศกดินแดง เขตดินแดง กรุงเทพฯ 10400 โทร. 248-6280-9 (อัตโนมัติ 20 สาย) แฟกซ์ 246-5999 โดยการส่งตั๋วแลกเงิน ธนาณัติหรือเช็คส่วนตัว สั่งจ่ายในนามบริษัทฯ กรณีเครดิตการ์ดโปรดระบุเงื่อนไขต่าง ๆ ตามการจ่ายเงินผ่านบัตรค่าจัดส่ง 8% พิเศษสำหรับสมาชิกวารสารของซีเอ็ด ลด 10% เพิ่มค่าลงทะเบียน 4 บาท ค่าจัดส่งฟรีโดยระบุหมายเลขสมาชิก